

In the Claims:

Please cancel claims 1-3, and add new claims 4-9, all as shown below.

1. – 3. (Canceled)

4. (New): A method for debugging in more than one programming language with a multi-language debugger, comprising:

providing an interface with a debugging frame for each language; and

allowing a user to edit each language in a debugging frame;

providing the capability to interpret multiple languages within a single source file and allow each of the multiple languages to be displayed in a debugging frame;

providing the ability to support additional languages; and

wherein the multi-language debugger uses a standardize interface for a script engine and all communications with the script engine will be through Java Debugging Interface calls to a script debug controller.

5. (New): The method of claim 4, wherein the first breakpoint behaves like a normal break and then the following process is performed, comprising the steps of:

a. the multi-language debugger receives the current class, line, and stack, and processes the stack through a language filter;

b. if the multi-language debugger encounters a class that implements a script language, the debug script controller will obtain the context and the contents of the stack;

c. script languages are processed as described in step (b), creating a stack frame list to send back to the multi-language debugger; and

d. the multi-language debugger then proceeds to discover and inspect variables in the same way as before.

6. (New): The method of claim 5, wherein the current stack frame is set to a frame controlled by a script engine, further comprising the steps of:
 - a. creating a list of variables for the current stack frame;
 - b. obtaining the values of the script variables through method invocations.
7. (New): The method of claim 5, wherein the multi-language debugger steps through code and informs the script debug controller when a step is about to begin.
8. (New): The method of claim 5, wherein the script engine supports JavaScript.